

Sub  
Pai 1  
2

- [illegible]



Sub  
Pat 1

- 1 12. The method of claim 11, wherein said executing utilizes a set of test data.
- 1 13. The method of claim 11, wherein said monitoring includes determining  
2 functions that consume a significant portion of said timing of said executing.
- 1 14. The method of claim 10, wherein said decomposing includes identifying  
2 kernel sections by identifying regular structures.
- 1 15. The method of claim 10, wherein said decomposing includes identifying  
2 kernel sections by identifying sections with a limited number of inputs and  
3 outputs.
- 1 16. The method of claim 10, wherein said decomposing includes identifying  
2 kernel sections by identifying sections with a limited number of branches.
- 1 17. The method of claim 10, wherein decomposing identifies overhead  
2 sections.
- 1 18. The method of claim 1, wherein mapping includes creating microcode.
- 1 19. The method of claim 1, wherein said mapping includes creating context  
2 dependent configurations.

1 20. The method of claim 1, wherein said matrix is sparsely-populated.

1 21. The method of claim 1, wherein said matrix is fully-populated.

1 22. A system for creating run time executable code, comprising:  
2 a plurality of hardware accelerators partitioned from a processing  
3 element array;  
4 a plurality of kernel sections created from a program description;  
5 a plurality of hardware dependent designs derived from said kernel  
6 sections; and  
7 a matrix describing said hardware accelerators and said designs  
8 configured to support run time execution.

1 23. The system of claim 22, wherein said hardware accelerators includes  
2 digital signal processors.

1 24. The system of claim 22, wherein said hardware accelerators includes  
2 bins.

1 25. The system of claim 24, wherein said bins support multiple hardware  
2 contexts.



1 33. The system of claim 32, wherein said software profiler includes a set of  
2 test data.

1 34. The system of claim 32, wherein said software profiler determines  
2 functions that consume a significant portion of said time consumed.

1 35. The system of claim 31, wherein said software profiler is configured to  
2 identify kernel sections by identifying regular structures.

1 36. The system of claim 31, wherein said software profiler is configured to  
2 identify kernel sections by identifying sections with a limited number of inputs  
3 and outputs.

1 37. The system of claim 31, wherein said software profiler is configured to  
2 identify kernel sections by identifying sections with a limited number of  
3 branches.

1 38. The system of claim 31, wherein said profiler identifies overhead  
2 sections.

1 39. The system of claim 22, wherein said designs include microcode.

1 40. The system of claim 39, wherein said microcode includes context  
2 dependent configurations.

1 41. The system of claim 22, wherein said matrix is sparsely-populated.

1 42. The system of claim 22, wherein said matrix is fully-populated.

1 43. A machine-readable medium having stored thereon instructions for  
2 processing elements, which when executed by said processing elements  
3 perform the following:

4 partitioning a processing element array into a plurality of hardware  
5 accelerators;

6 decomposing a program description into a plurality of kernel sections;

7 mapping said kernel sections into a plurality of hardware dependent  
8 designs; and

9 forming a matrix describing said hardware accelerators and said designs  
10 configured to support run time execution.

Sub  
Pat

1  
2  
3  
4  
5  
6  
7  
8  
9

000000-000000

44. A system configured to create run time executable code, comprising:
- means for partitioning a processing element array into a plurality of hardware accelerators;
  - means for decomposing a program description into a plurality of kernel sections;
  - means for mapping said kernel sections into a plurality of hardware dependent designs; and
  - means for forming a matrix describing said hardware accelerators and said designs configured to support run time execution.